DEPARTMENT: AFFECTIVE COMPUTING AND SENTIMENT ANALYSIS

No, That Never Happened!! Investigating Rumors on Twitter

Md Shad Akhtar
Asif Ekbal
Sunny Narayan
Vikram Singh
Indian Institute of Technology Patna

Editor:
Erik Cambria,
cambria@ntu.edu.sg

Rumors are the statements which cannot be verified for correctness. Intuitively, a fact statement cannot be a rumor or vice-versa. The spread of misinformation is, especially important in the context of breaking news, often starting off as unverified information in the form of a rumor. These rumors then spread to a large number of users, influencing perception and understanding of events, despite being unverified. Misinformation often creates a confusing and chaotic scenario, which leads to poor decision-making or many inhuman consequences. Social media rumors that are later proven false can have harmful consequences both for individuals and for society. For instance, a rumor in 2013 about the White House having been bombed, injuring Barack Obama, spooked stock markets in the United States. In another instance, rumors played a vital role in deadly 2011 London riots. Rumor detection and its support invite great interest from various organizations and government agencies. By keeping track of such information, they can take preventative measures to maintain good decorum. Therefore, determining the authenticity of the circulating misinformation (rumors) in a timely manner is very crucial.

In this paper, we focus on two problems related to rumor detection. First, stance classification w.r.t. a rumor and, second, rumor veracity prediction. In the stance classification task, we aim to identify the users' stance toward the underlying rumor in a Twitter conversational thread, whereas, in the second problem, i.e., veracity prediction, we aim to verify the authenticity of a rumorous tweet (i.e., source tweet) in a conversational thread.

We realize that capturing the users' intentions in a tweet is not a straightforward task. Tweets can have a mixture of stances; for example, "*@hking u r right that 10 people were killed but no bomb blast happened #terroristAttack*," contains two stances, support and denial, by supporting the fact about the number of people killed but also denying the fact about the bomb blast. The veracity of any rumor relies heavily on the world knowledge. In the absence of it, we cannot be sure about the truthfulness of the reported incidents. At the preliminary step, user stance toward a reported incident (rumor or nonrumor) may assist in establishing the veracity of the rumor. An example scenario is given in Table 1.

The conversational thread discusses an underlying rumor in the source tweet and for which other users have expressed their views in replies and nested replies. Each of these views poses a relation with the

Published by the IEEE Computer Society
1541-1672/18/$33.00 ©2018 IEEE

Table 1. Tree structured conversation thread in Twitter. Source tweet is a rumor.

Source: Police now say there were two shooting incidents in Ottawa: one at the war memorial, the other on Parliament Hill. URL

- **R1:** @cnnbrk: No, that never happened!!—**deny**
- **R2:** @cnnbrk: Now two shooting incidents in Ottawa: one at the war memorial, the other on Parliament Hill. URL #scary—**support**
- **R3:** @cnnbrk Weren't you reporting THREE incidents earlier?—**query**
  - **R4:** @ViewFromAskew @cnnbrk breaking news is fraught with mistakes. Calm down.—**deny**
    - **R5:** @palescalesDRUNK @cnnbrk - You mean like someone who harps on someone else asking for clarification? Sounds like *you* need to calm down.—**comment**
- **R6:** @cnnbrk: Police now say there were 2 shooting incidents in Ottawa: at the war memorial, and on Parliament Hill. URL—**comment**

underlying rumor in terms of the four classes—*support, deny, query*, and *comment*. The *support* class voted for the veracity of the rumor, whereas *deny* rejects it. In *comment*, users express their own perception without a clear contribution in assessing the veracity. Some users ask for additional information/evidence about the incident in the form of *query*. For example, the first reply (*R1: @cnnbrk: No, that never happened!!*) of the conversational thread in Table 2 is denying the fact that some shooting incidents have happened, whereas reply *R2* is supporting the incident.

RELATED WORKS

A survey of the literature suggests detecting stance for a rumor has gained lots of attention lately. Earlier work on rumor stance detection[1] treated the problem of long-standing rumor as a two-class classification problem, i.e., support or deny. Liu *et al.*[2] defined a rule-based method to improve the performance of Qazvinian *et al.*[1] More recent works (including ours) target rumors that emerge with the *breaking-news*.[3–5] Lukasik *et al.*[3] utilized a Gaussian process-based multitasking learning technique trained on BoW/Brown cluster ids for predicting the stance of a tweet during the 2011 England riots. Application of Hawkes process[6] has been studied[4] to exploit the temporal sequence of stances toward rumors. Similarly, Zubiaga *et al.*[5] modeled tree-structure in a conversational thread using a conditional random field for the stance classification.

Srivastava *et al.*[7] proposed a hybrid model that utilized a multivariate logistic regression-based machine learning technique and a set of heuristics in cascading for the stance detection in rumor and rumor veracity prediction. An application of a convolutional neural network (CNN) for both the problems was proposed by Chen *et al.*[8] Enayet and El-Beltagy[9] proposed a linear SVM-based classifier learned on top of various linguistics, user and twitter-specific features. Additionally, they manually selected an optimal feature set through cross-validation. Similarly, Singh *et al.*[10] proposed a supervised approach for training SVM and a Naïve Bayes (NB) classifier for stance detection and veracity prediction, respectively.

The system reported in Wang *et al.*[11] adapts a series of classifiers over a diverse set of features and then combines the prediction through a majority voting scheme for detecting the stance and establishing veracity. The work reported by Bahuleyan and Vechtomova[12] exploits the presence of various cue words along with the tweet-specific features to train a gradient boost classifier for predicting the veracity of a rumor. Kochkina *et al.*[13] proposed an LSTM-based deep learning network to establish the veracity of a rumor. Chen *et al.*[14] proposed an acquaintance network that can be adapted for the research on rumor propagation. Further, the application of user profiling or personality detection[15] can also assist the rumor veracity prediction system by validating the source of a potential rumorous tweet.

Although emotion and sentiment analysis[16–19] can play an effective role in identifying rumors, it offers a different dimension as we are not only interested in knowing the individual's opinion or sentiment, but also its impact and/or influences on the society at large.

PROPOSED METHODOLOGY

In subsequent subsections, we describe our proposed approach for rumor stance detection and veracity prediction problems.

Stance Classification in Rumor

For detecting the stance of a tweet w.r.t. the underlying rumor, we propose to exploit the thread structure of a conversation. To tag a reply tweet *R*, we first create a tweet chain comprised of the source tweet and all other replies in the path from the source tweet through tweet *R*. Thus, the instance contains the sequential thread structure of the tweet, which we aim to utilize for classification. Following this step, we create instances for each reply in the thread. For each instance, we aim to classify the last tweet in the chain, which essentially depends on its parent tweet and so on. For the example given in Table 1, we create six instances, as shown in the following.

Instance 1: Source, **R1**	**deny**
Instance 2: Source, **R2**	**support**
Instance 3: Source, R3	**query**
Instance 4: Source, R3, **R4**	**deny**
Instance 5: Source, R3, R4, **R5**	**comment**
Instance 6: Source, **R6**	**comment**

In the proposed architecture, i.e., hierarchical long short term memory (LSTM) network, at first we learn sentence embedding of each tweet through the first layer LSTM, and then the learned embedding is fed into the second layer LSTM network to exploit the sequence information of the reply chain. Figure 1 depicts the proposed architecture for the hierarchical LSTM.

Rumor Veracity Prediction

Unlike the stance detection task, we adopt a supervised feature-driven approach for establishing the veracity of a rumor. The choice of a feature-driven approach as compared to a deep learning technique was due to an insufficient number of instances in the benchmark datasets of SemEval-2017 shared task on rumoreval[20] that we utilized for training and evaluation in this paper.

We define and extract a diverse set of features for learning the classifier. We employ support vector machine (SVM), NB, decision tree (DT), and multilayer perception (MLP) as our classification algorithms.

Feature Set

We implement the following features for predicting the veracity of a rumor.

I. *Word embeddings*: We use a 200-dimension pretrained GloVe for computing the word embeddings.

II. *Subjectivity information*: A statement that carries opinions (e.g., positive, negative, etc.) is known as a subjective statement, whereas an objective statement (fact) does not contain any opinion. As discussed earlier, a fact statement cannot be a rumor. In other words, a sentiment-bearing statement is a possible candidate for being a rumor. We identify and use the following set of features for extracting the subjectivity information of a tweet.

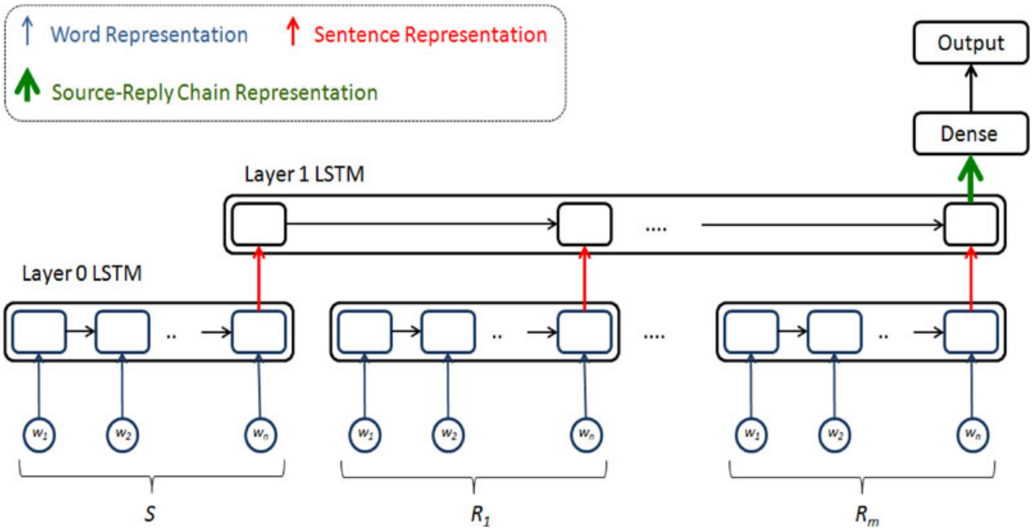

Figure 1. Proposed methodology: Hierarchical LSTM architecture.

A. *NLTK sentiment score*: We use an NLTK sentiment analysis package to extract the polarity information (i.e., *positive, negative,* and *neutral*) of each tweet.

B. *Subjectivity score*: We define two features based on BingLiu opinion lexicon to extract the subjectivity information. The first feature counts the number of opinion words in a tweet, whereas the second feature identifies the presence/absence of a strong subjective word in a tweet.

C. *Vulgar words*: The possibility of the presence of a vulgar or slang word in a fact statement is very low, thus making it a probably candidate for a rumorous statement.

D. *Presence of adjectives*: The presence of adjectives in a sentence has a strong relationship with being a subjective statement. We define a feature that marks the presence of adjectives in a sentence.

E. *Temporal information*: The presence of temporal information in a statement signifies it being a factual statement (more than an opinion). We define and use a feature that identifies the presence of temporal information in a tweet.

III. *Word count*: Rumors are generally longer and more elaborative, whereas factual statements are normally short and precise. We take word count (excluding stop words) as a feature for building the system.

IV. *Twitter-specific features*:

A. *Metadata*: The presence of metadata, e.g., URLs, Media, etc., in a tweet indicates that the statement is supported by authentic information. However, rumorous statements do not have such privilege due to the absence of authentic sources. We implement a binary feature that fires when some metadata is available with the tweet.

B. *Retweet (RT)*: Rumors spread very fast within a short burst of time, hence they usually have an high RT count. A value indicating the RT count of a tweet is taken as a feature value in this paper.

C. *Punctuation, emoticons, and abbreviations.*

V. *User-specific features*: Rumor mongers usually spread false information through anonymity and do not wish to reveal their identity, whereas the probability of a reliable source spreading a rumor is extremely low. We define the following set of features to capture the reliability of a twitter user.

A. *Account creation date*: Rumors are usually started from newly created accounts. An individual (or organization) who is a longtime Twitter user is less likely to spread the false propaganda/rumors. We define a binary feature that takes the value 1 if the account was created more than 30 days before the tweet date; otherwise 0.

B. *Verified user account*: An account which is unverified would most likely be involved in spreading rumorous stories compared to the verified accounts.
C. *Profile picture*: To hide their identity from the investigation agency/police, rumor mongers do not use their profile picture, whereas a common user does not have such concerns.
D. *Geographic location information*: The geographic location feature in Twitter indicates the whereabouts of a user. Those who spread rumors do not want their location to be traced, hence usually have their geographic-location feature disabled.
E. *Follower counts*: Follower count reflects the popularity of a user among others. It is very unlikely that such a renowned user would spread false information. In contrast, a user who spread misinformation would rarely get a high follower count.

EXPERIMENTS, RESULTS, AND ANALYSIS

Dataset

For evaluation purposes we use the datasets of a SemEval-2017 shared task on "RumourEval: Determining rumour veracity and support for rumours."[20] The dataset is associated with eight events corresponding to the breaking news stories circulated over the Internet and news. Tweets in the dataset include tree-structured conversations, which are initiated by a potential rumorous tweet (source tweet). Other users on Twitter expressed their views through replies and nested replies, thus creating a conversation thread. Each of these replies and nested replies might have different stances (*support, deny, comment*, and *query*) toward the source tweet (potential rumor). The potential rumor can be *true, false,* or *unverified*. The dataset consists of 3966, 256, and 1021 replies across 272, 25, and 28 conversational threads discussing potential rumors in the training, development, and test set, respectively.

Experiments

We use Python-based libraries Keras and Scikit-learn for the implementations of stance classification and veracity prediction. Following the guidelines of the SemEval-2017 shared task on "RumourEval,"[20] we use accuracy value as a metric for the stance classification and macro-averaged accuracy for the veracity prediction.

For the stance detection problem, we first train a vanilla LSTM architecture consisting of two LSTM layers followed by two dense layers and one output layer. In the vanilla LSTM architecture, we concatenate word embeddings of each token to form a tweet and then feed it to the network for classification. This architecture reports an accuracy of 0.7861. We further experiment with the hierarchical architecture. In the first layer of the network, we learn the sentence embeddings through an LSTM layer, which is then fed to the second layer of LSTM. The second layer learns the stance over the sentence embedding of source and reply tweets at each time step. The hierarchical architecture

Table 2. Feature ablation for rumor veracity prediction.

Features	Veracity prediction (Macro-avg accuracy)			
	NB	**DT**	**SVM**	**MLP**
All	0.286	0.286	0.571	0.646
-User-specific features	0.286	0.357	0.536	0.571
-Subjectivity features	0.393	0.464	0.428	0.464
-Twitter-specific features	0.286	0.286	0.286	0.286

Table 3. Stance detection and veracity prediction: Performance of the proposed system and various state-of-the-art systems.

Systems	Descriptions	Stance detection	Veracity prediction
Baseline[20]	Most common class	0.741	0.571
DFKI-DKT[7]	Multivariate LR + Heuristics	0.635	0.393
IITP[10]	Features—SVM/NB	0.640	0.286
IKM[8]	CNN	0.701	0.536
NileTMRG[9]	Manually optim. features	0.709	0.536
ECNU[11]	Ensemble—Majority Voting	0.778	0.464
UWaterloo[12]	Features—Gradient boost	0.780	–
Turing[13]	Vanilla LSTM	0.784	–
Proposed method	**Vanilla LSTM**	**0.786**	–
	Hierarchical LSTM	**0.799**	–
	Features—MLP		**0.642**

reports an improvement over the vanilla architecture at 0.7988 accuracy value. The improvement can be attributed to the learned sentence embeddings.

Due to a small-sized dataset, rather than using deep learning models, we train four classical supervise models with a set of features (cf. Section *Feature Set*) for the veracity prediction. Results reported in Table 2 show that MLP obtains the best macro-average accuracy of 0.642.

We also study the importance of different subsets of feature combinations through a feature ablation approach, and its results are given in Table 2. NB reports the best macro-average accuracy of 0.393 when no user-specific and subjectivity features are used. We observe a similar case for DT where it attains a macro-average accuracy value of 0.464 without user-specific and subjectivity features. However, both SVM- and MLP-based classifiers attain an improved macro-average accuracy value of 0.571 and 0.642, respectively, when the complete feature set is utilized. We observe that without user-specific, subjectivity, and twitter-specific features, our system obtains a 0.286 macro-average accuracy value for all four classifiers.

We further compare the performance of our proposed system with that of the various state-of-the-art systems.[7–13] A detailed comparison of the proposed approach with various state-of-the-art and baseline systems are reported in Table 2 for both the stance classification and the veracity prediction problems.

For stance detection, Turing[13] and UWaterloo[12] are the top two systems at the SemEval-2017 shared task on rumoreval[20] with accuracy values of 0.784 and 0.780, respectively. In comparison, our proposed system attains an improved accuracy value of 0.799. The system reported by Bahuleyan and Vechtomova[12] trained a gradient boost classifier on top of twitter-specific and cue word features for the predictions, whereas the Turing[13] employed an LSTM architecture similar to the vanilla architecture with the exception of classifying each tweet in the structure at each time step. However, our proposed system classifies the last tweet of each instance separately. Also, we learn sentence embeddings of each tweet through an LSTM layer rather than just concatenating the embedding of each token of the tweet.

In the veracity prediction problem, IKM[8] and NileTMRG[9] are the top two systems at SemEval-2017 shared task on rumoreval.[20] IKM[8] proposed a CNN for the predictions, whereas NileTMRG[9] adopts a linear SVM classifier to learn on top of a manually optimized feature set. Both of these systems report similar performances at 0.536 macro-average accuracy. It should be noted that the baseline system for veracity prediction as reported[20] performs better than each of the existing systems. In contrast, our proposed approach reports an improved macro-average accuracy of 0.642, which is an improvement of approximately 11 points over the best reported system and approximately 7 points over the baseline system.

We also perform a statistical significance test on the obtained results and observe that the performances that we obtain are significant with p value $= 0.019218$ and p value $= 0.042689$ for the stance classification and veracity prediction, respectively.

Error Analysis

Further, we carry out error analysis on the obtained results. Since the dataset is biased toward the *comment* class, our proposed system has high recall (96.5%) but relatively low precision (80.2%) for *comment*. The most problematic class was *deny*, which reports 100% precision but a merely 1.4% recall. Out of 71 instances, 67 instances were tagged as *comment*. On further analysis, we observe that most of these instances do not reject the statement directly; rather, they were implicit, sarcastic, or ironic, etc. An example scenario is listed in the following. All the replies are inherently denying that "*marina joyce has a drug addiction*," but the system tagged all of them as *comment* due to the absence of any explicit denial.

Rumorous tweet: *Anyone who know marina joyce personally knows she has a serious drug addiction. She needs help, but in the form of rehab #savemarinajoyce*

> **Reply 1**: *@user who are u*
> **Reply 2**: *@user please, quit the internet*
> **Reply 3**: *@user, you realllyy have no place to talk.*

CONCLUSION

Rumor analysis in social media text has vast applications in real-world scenarios. Detecting rumor veracity and user's stance on an underlying rumor are the two basic problems of rumor analysis. In this paper, we have proposed an MLP-based feature-driven model for veracity prediction and a hierarchical LSTM-based approach for detecting stances toward a rumor in a conversation thread. Evaluations show that our proposed system attained better performance in comparison with the various state-of-the-art systems on both the tasks.

REFERENCES

1. V. Qazvinian, E. Rosengren, D. R. Radev, and Q. Mei., "Rumor has it: Identifying misinformation in microblogs," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2011, pp. 1589–1599.
2. X. Liu, A. Nourbakhsh, Q. Li, R. Fang, and S. Shah, "Real-time rumor debunking on twitter," in *Proc. 24th ACM Int. Conf. Inf. Knowl. Manage.*, 2015, pp. 1867–1870.
3. M. Lukasik, T. Cohn, and K. Bontcheva, "Classifying tweet level judgments of rumours in social media," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 2590–2595.
4. M. Lukasik, P. K. Srijith, D. Vu, K. Bontcheva, A. Zubiaga, and T. Cohn, "Hawkes processes for continuous time sequence classification: An application to rumour stance classification in twitter," in *Proc. 54th Annu. Meet. Assoc. Comput. Linguistics: Short Papers*, 2016, pp. 393–398.
5. A. Zubiaga, E. Kochkina, M. Liakata, R. Procter, and M. Lukasik, "Stance classification in rumours as a sequential task exploiting the tree structure of social media conversations," in *Proc. 26th Int. Conf. Comput. Linguistics: Tech. Papers*, 2016, pp. 2438–2448.
6. A. G. Hawkes, "Spectra of some self-exciting and mutually exciting point processes," *Biometrika*, vol. 33, no. 50, pp. 83–90, 1971.

7. A. K. Srivastava, G. Rehm, and J. M. Schneider, "DFKI-DKT at SemEval-2017 Task 8: Rumour detection and classification using cascading heuristics," in *Proc. 11th Int. Workshop Semantic Eval.*, 2017, pp. 486–490.
8. Y.-C. Chen, Z.-Y. Liu, and H.-Y. Kao, "IKM at SemEval-2017 Task 8: Convolutional neural networks for stance detection and rumor verification," in *Proc. 11th Int. Workshop Semantic Eval.*, 2017, pp. 465–469.
9. O. Enayet and S. R. El-Beltagy, "NileTMRG at SemEval-2017 Task 8: Determining rumour and veracity support for rumours on twitter," in *Proc. 11th Int. Workshop Semantic Eval.*, 2017, pp. 470–474.
10. V. Singh, S. Narayan, M. S. Akhtar, A. Ekbal, and P. Bhattacharyya, "IITP at SemEval-2017 Task 8: A supervised approach for rumour evaluation," in *Proc. 11th Int. Workshop Semantic Eval.*, 2017, pp. 497–501.
11. F. Wang, M. Lan, and Y. Wu, "ECNU at SemEval-2017 Task 8: Rumour evaluation using effective features and supervised ensemble models," in *Proc. 11th Int. Workshop Semantic Eval.*, 2017, pp. 491–496.
12. H. Bahuleyan and O. Vechtomova, "UWaterloo at SemEval-2017 Task 8: Detecting stance towards rumours with topic independent features," in *Proc. 11th Int. Workshop Semantic Eval.*, 2017, pp. 461–464.
13. E. Kochkina, M. Liakata, and I. Augenstein, "Turing at SemEval-2017 Task 8: Sequential approach to rumour stance classification with Branch-LSTM," *in Proc. 11th Int. Workshop Semantic Eval.*, 2017, pp. 475–480.
14. X. Chen, L. Zhang, and W. Li, "A network evolution model for Chinese traditional acquaintance networks," *IEEE Intell. Syst.*, vol. 29, no. 5, pp. 5–13, Sep./Oct. 2014.
15. N. Majumder, S. Poria, A. Gelbukh, and E. Cambria, "Deep learning-based document modeling for personality detection from text," *IEEE Intell. Syst.*, vol. 32, no. 2, pp. 74–79, Mar./Apr. 2017.
16. E. Cambria, B. Schuller, Y. Xia, and C. Havasi, "New avenues in opinion mining and sentiment analysis," *IEEE Intell. Syst.*, vol. 28, no. 2, pp. 15–21, Mar./Apr. 2013.
17. E. Cambria, "Affective computing and sentiment analysis," *IEEE Intell. Syst.*, vol. 31, no. 2, pp. 102–107, Mar./Apr. 2016.
18. E. Cambria, S. Poria, A. Gelbukh, and M. Thelwall, "Sentiment analysis is a big suitcase," *IEEE Intell. Syst.*, vol. 32, no. 6, pp. 74–80, Nov./Dec. 2017.
19. M. Ebrahimi, A. H. Yazdavar, and A. Sheth, "Challenges of sentiment analysis for dynamic events," *IEEE Intell. Syst.*, vol. 32, no. 5, pp. 70–75, Sep./Oct. 2017.
20. L. Derczynski, K. Bontcheva, M. Liakata, R. Procter, G. W. S. Hoi, and A. Zubiaga, "SemEval-2017 Task 8: RumourEval: Determining rumour veracity and support for rumours," in *Proc. 11th Int. Workshop Semantic Eval.*, 2017, pp. 69–76.

ABOUT THE AUTHORS

Md Shad Akhtar is a research scholar with the Department of Computer Science and Engineering, IIT Patna. He has more than two years of working experience with HCL Tech Ltd., New Delhi. His research interests include natural language processing and sentiment analysis. He received the M.Tech degree from the IIT (ISM), Dhanbad, in 2014. He has authored or co-authored various peer reviewed conferences and journals of international repute. More information can be found at http://www.iitp.ac.in/~shad.pcs15/index.html. Contact him at shad.pcs15@iitp.ac.in.

Asif Ekbal is currently an associate professor with the Department of Computer Science and Engineering, IIT Patna. He is involved with different sponsored research projects in the broad areas of artificial intelligence and machine learning technologies, funded by different government and private agencies. His research interests include natural language processing, information extraction, text mining, and machine learning applications for the last 11 years. He has authored around 150 papers. More information can be found at: http://www.iitp.ac.in/~asif/index.html. Contact him at asif@iitp.ac.in.

Sunny Narayan received the B.Tech degree in computer science engineering from the IIT Patna. Contact him at sunny.cs13@iitp.ac.in.

Vikram Singh received the M.Tech degree in computer science engineering from the IIT Patna. Contact him at vikram.mtcs15@iitp.ac.in.